Jitter Measurements in the 0.7 4.0 PCI Express® Base Specification

Savitha Muthanna
Software Design Engineer
Tektronix

Disclaimer

<u>Presentation Disclaimer</u>: All opinions, judgments, recommendations, etc. that are presented herein are the opinions of the presenter of the material and do not necessarily reflect the opinions of the PCI-SIG®.

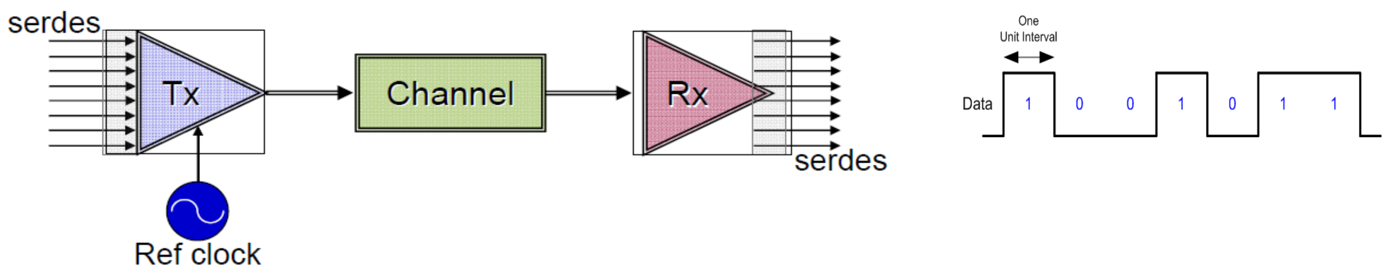
The information in this presentation refers to specifications still in the development process. Material is subject to change before the specifications are released.

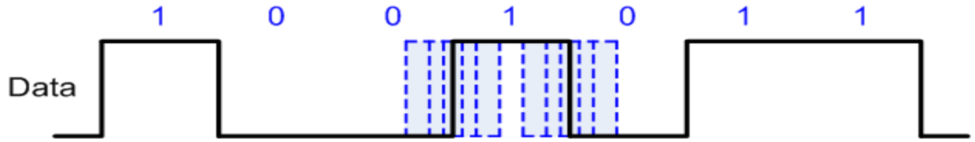
Agenda

- Introduction
- New, 2.5GT/s and 5GT/s Jitter Measurements
- Reference Clock Jitter Measurements
- Conclusion
- References

Jitter - Introduction

- **Jitter is formally defined as short-term variations of the significant instants of a digital signal from their ideal positions in time.**





Jitter Components in the PCI™ Spec

- **Data-dependent jitter**
  - Package loss, reflection
  - Non-budgeted, and recoverable
- **Uncorrelated jitter**
  - PLL jitter, power supply noise, and crosstalk
  - Budgeted and non-recoverable
  - Divisible into Total Jitter(TJ) and Deterministic Jitter(DJ)

New Jitter Measurements for 2.5GT/s and 5GT/s

- Uncorrelated Total Jitter (UTJ)/Uncorrelated Deterministic Jitter Dual Dirac (UDJDD) for 2.5GT/s and 5GT/s
- Uncorrelated Pulse Width Total Jitter(UPW-TJ)/Uncorrelated Pulse Width Deterministic Jitter Dual Dirac(UPW-DJDD) for 2.5GT/s and 5GT/s
  - PCIe® 3.0 methodology is applied to 2.5GT/s and 5GT/s, except pattern matching, to unify the standard

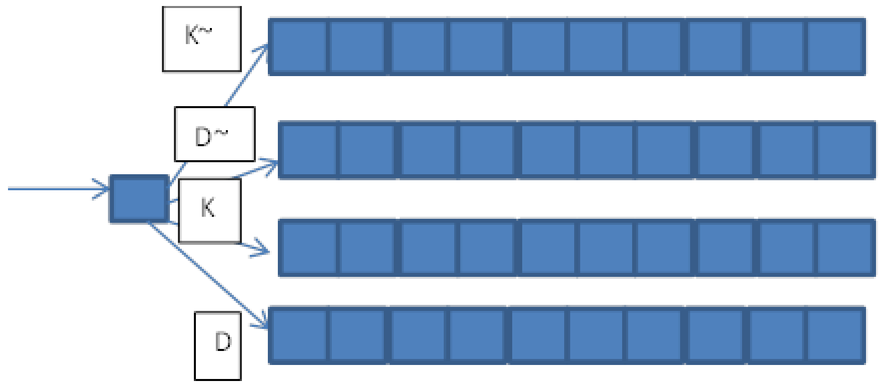
Symbol	2.5 GT/s	5.0 GT/s
TTX-UTJ	100 (max)	50(max)
TTX-UDJDD	100(max)	30(max)
TTX-UPW-TJ	N/A	40(max)
TTX-UPWDJDD	N/A	40(max)

Computation of UTJ/UDJDD

- Acquire a sufficiently large record of repetitions of the 8b/10b compliance pattern
- Detect edges
- Calculate the data dependent jitter (DDJ)
- Construct a PDF (probability density function) of edges with only the uncorrelated component.
- Convert to Q-scale
- Derive UTJ/UDJDD

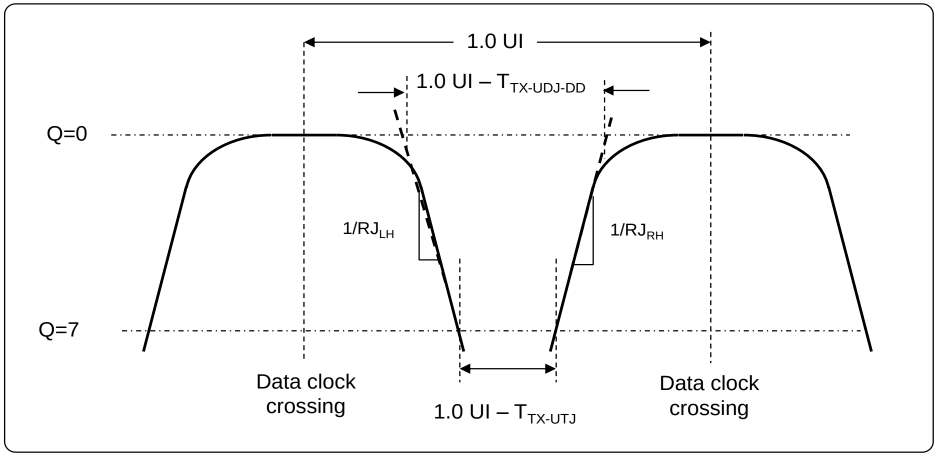
Computation of DDJ

- The compliance pattern for 2.5GT/s and 5GT/s is the repeating K and D patterns of the 8b/10b encoding scheme [2] and their complements
- Search for the first 10-bit pattern(K/D)
- Find the second 10-bit pattern
- Using regular expression match, K/D pattern occurrences are instantaneously identified
- A dictionary is then used to store the index of the edges of the corresponding patterns



- The correlated jitter is then calculated as the mean of the accumulated edges for each bit in each pattern
- This mean is eliminated from each unique edge to remove the data dependent jitter and leave only the uncorrelated jitter component

Computation of UTJ/UDJDD (cont'd)

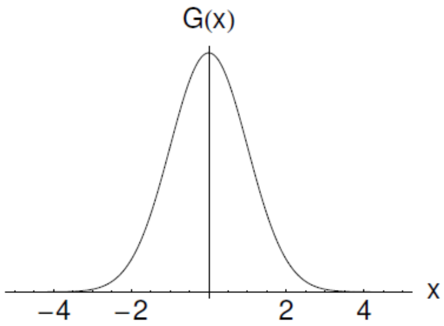


- **Construct a PDF(probability density function) with edges with only the uncorrelated component**
  - Pick an optimum bin size
  - Identify min and max values
  - Construct the PDF

Probability Density Functions

- **A note about probability density functions:**
  - Probability Density Function (PDF), or density of a continuous random variable, is a function that describes the relative likelihood for this random variable to take on a given value.
  - Histograms are closest in terms of measurable probability density, and are said to approximate the PDF when normalized.
  - The probability density function for the Gaussian distribution is given by:

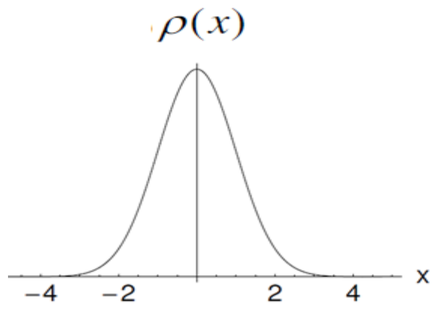
$$f_X(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{(x-\mu)^2}{2\sigma^2}\right], \quad -\infty < x < \infty.$$

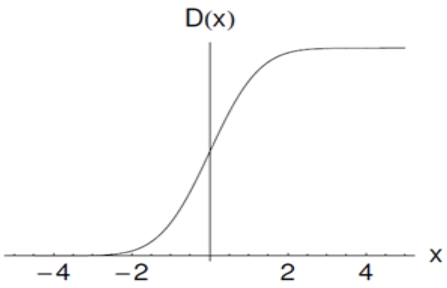


Computation of UTJ/UDJDD (cont'd)

- **The PDF is converted to CDFs by cumulative summing.**
  - CDF stands for cumulative distribution function. The CDF of a random variable X, can be expressed as the integral of its probability density function. It is the probability that X will take a value less than or equal to x. It is also called the Right CDF. The Complementary CDF or left CDF is integrated from ∞ to x.

$$CDF(x) = \int_{-\infty}^{x} PDF(u)du$$





Computation of UTJ/UDJDD (cont'd)

Convert to Q-scale by applying the error function.

- The error function is obtained by integrating the normalized Gaussian distribution

$$G(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-x^2/(2\sigma^2)}$$

$$\int_{-x}^{x} G(x)\, dx = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2}\, dt = \operatorname{erf} x$$

The relationship between CDF and error function is expressed as:

$$CDF(x) = \frac{1}{2} + \frac{1}{2} erf \left[\frac{x}{\sigma\sqrt{2}}\right]$$

The Q scale transformation of a Gaussian CDF is given as follows:

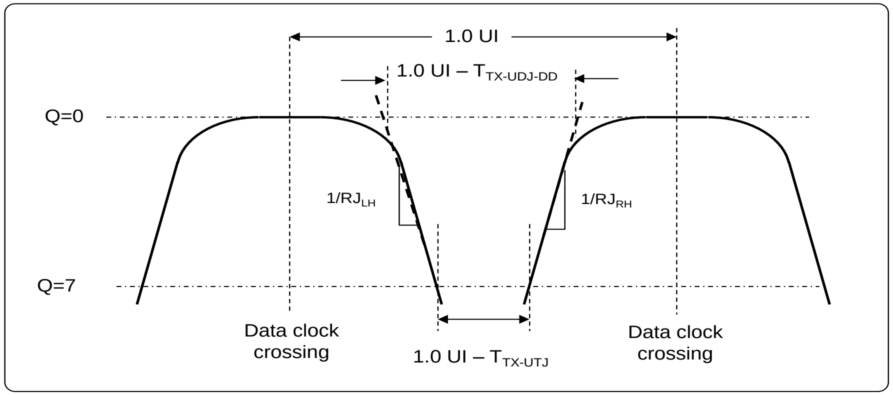
$$Q(x) = \sqrt{2} erf^{-1}(2CDF(x) - 1) = \frac{x}{\sigma}$$

Therefore, a Gaussian CDF is a straight line in the Q scale, with slope one over sigma;

Computation of UTJ/UDJDD (cont'd)

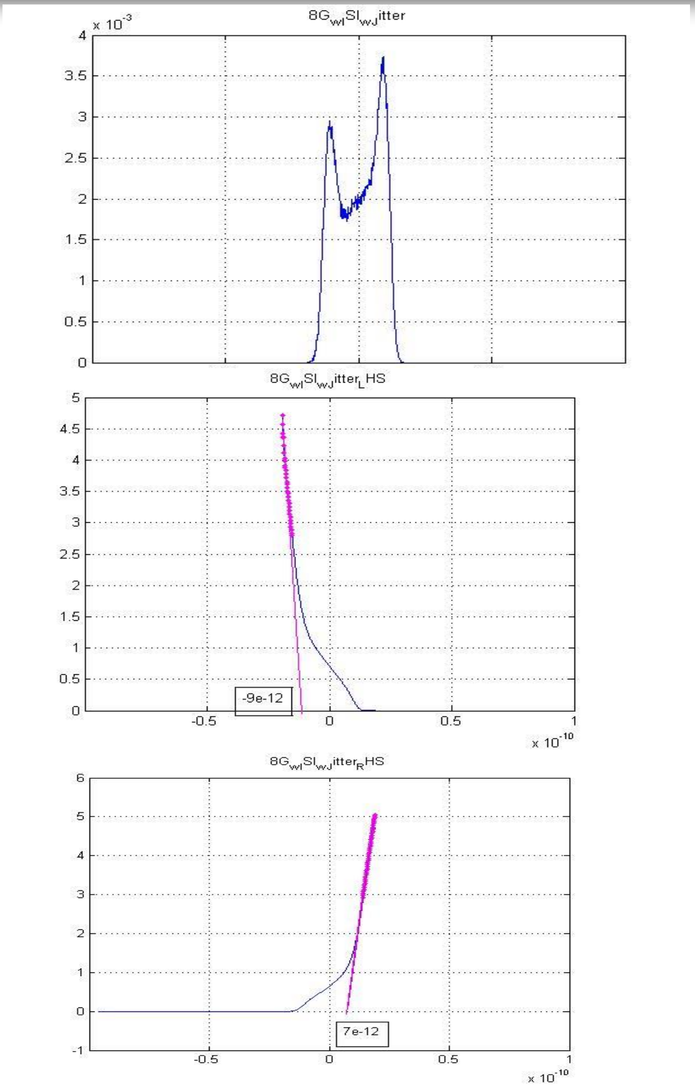
Calculate UDJDD

- For calculating the UDJDD, we calculate the slope by taking the data points on the tail of the Q scale and estimate the intersection at Q=0 by extrapolating.
- 1UI **-** UDJDD value is the distance between two intercepts (RHS-LHS), from which UDJDD is calculated. This refers to the eye width.



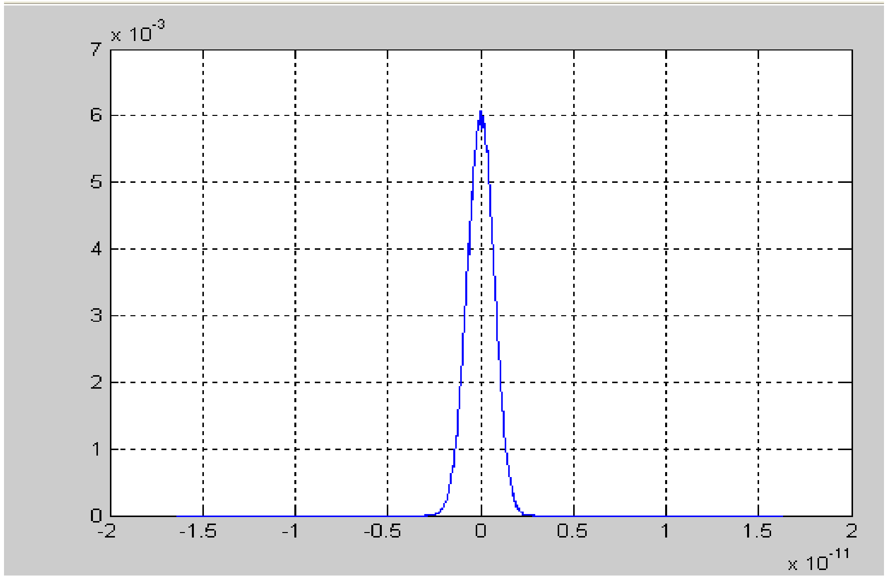
Computation of UTJ/UDJDD (cont'd)

- **Illustration of UDJDD measurement:**
  - Histogram after removal of DDJ.
  - Convert the PDF to LCDF(Left CDF) and then to Q scale
  - Estimate the slope by taking data point from the 2.5 level and extend to zero for LHS
  - Similarly take the RCDF(Right CDF), convert to Q scale, and extend to zero for RHS
  - LHS is -9 psec, RHS is 7 psec
    UDJDD = 16 psec



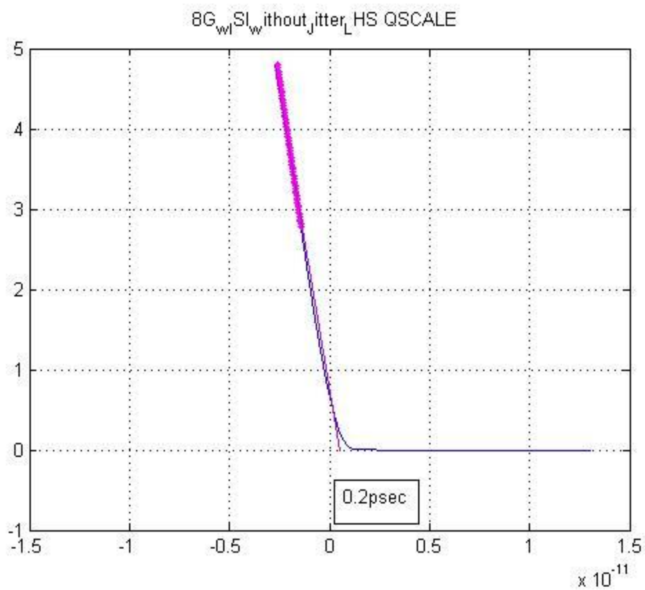
Sources of Error - Computation of UTJ/UDJDD

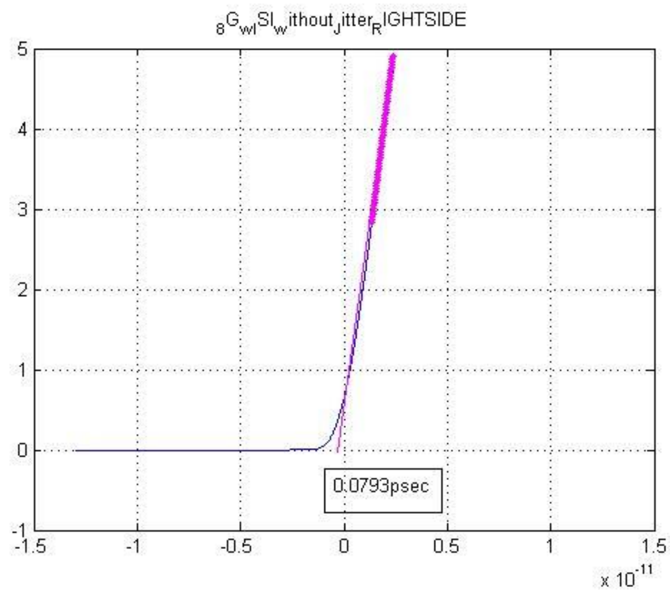
- **Sometimes UDJDD is measured as negative:**
  - This could be because Rj is not really Gaussian, but the necessity of using a Gaussian model could cause it.



Sources of Error - Computation of UTJ/UDJDD

Sometimes UDJDD is measured as negative (cont'd):





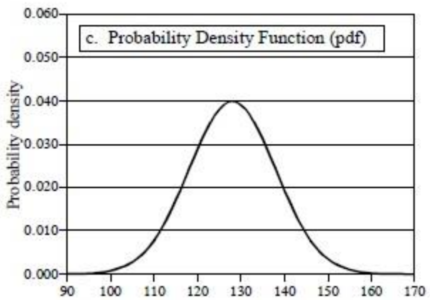
LHS intersects on Right side of X axis(Quad 1) RHS intersects on Left side of X axis (Quad2) DJ of -0.0793-0.2 = -279 fs.

Sources of Error - Computation of UTJ/UDJDD

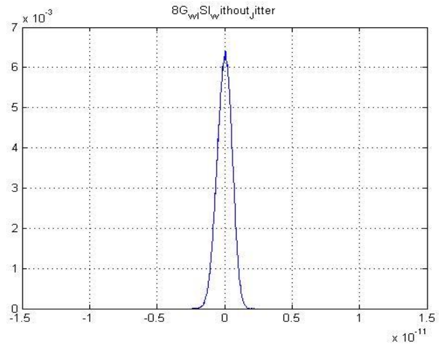
- **Sometimes UDJDD is measured as negative (cont'd):**
  - The estimation of the slope can be improved by locating the Gaussian start point on the tail of the histogram. The inflection point was arrived at experimentally from actual PCIe DUTs.
  - A note about tail points: Points near the tail, will have a bin value above a threshold hit count, which is arrived based on the population and max hit count. By doing this we get the tail portion that is concrete enough to give the consistent measurement results.

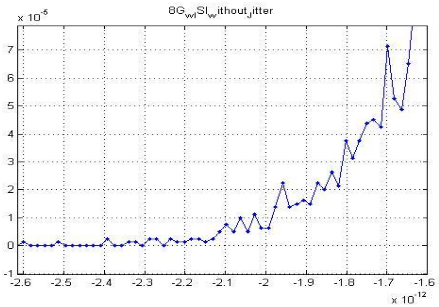
Sources of Error - Computation of UTJ/UDJDD

The model expects Monotonicity:



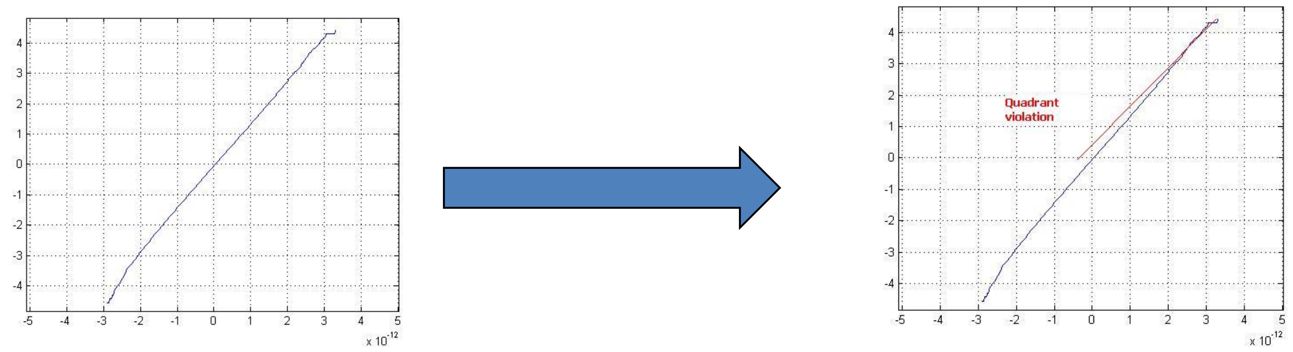
Practically histogram is discrete, Not all the bins gets filled, and may not exhibit monotonicity.



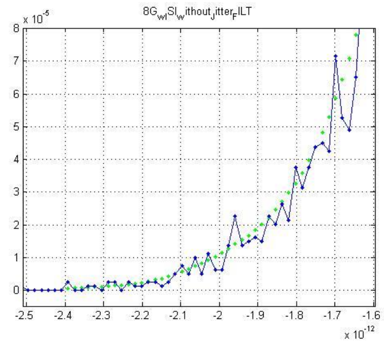


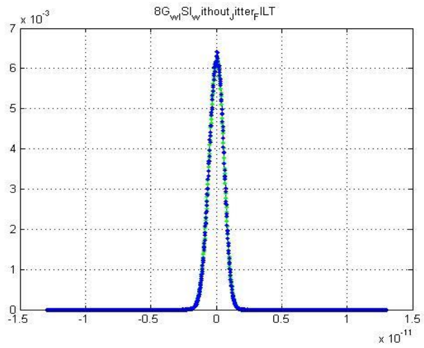
Sources of Error - Computation of UTJ/UDJDD(cont’d)

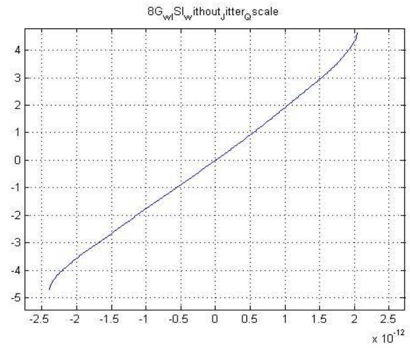
If some of the bins are zero, when we extend the slope of DJ it may violate quadrants.



We filter the histogram bin data to ensure monotonicity, so that there is no phase shift for the given bin and the jitter is unaltered.

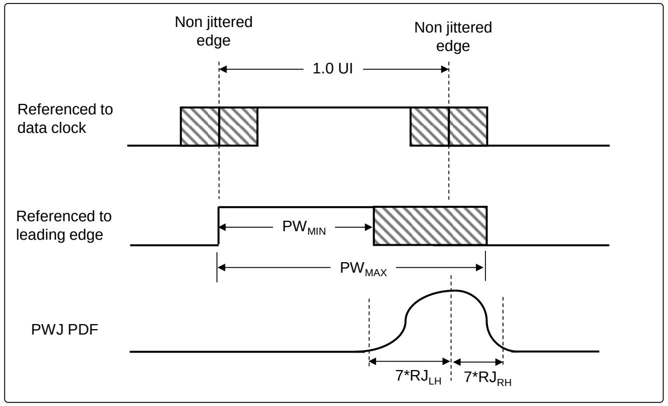






Computation of UPW-TJ/UPW-UDJDD

The Pulse Width jitter is defined on pulses, 1 UI apart, typically present on both data edges of consecutive UI.

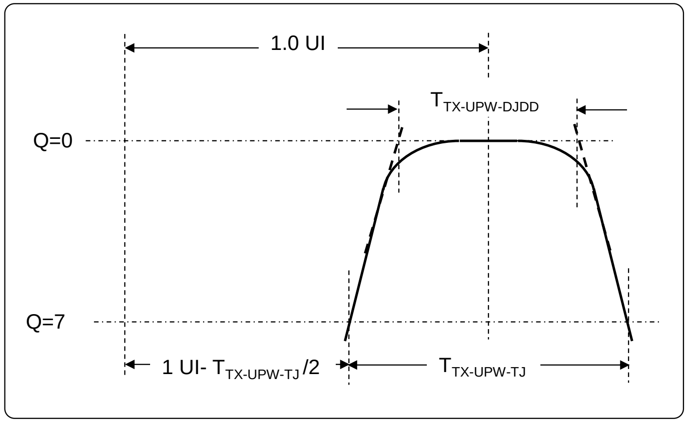


Computation of UPW-TJ/UPW-UDJDD

- **Pulse Width Jitter is calculated as follows:**
  - Acquire a sufficiently large record of repetitions of the 8b/10b compliance pattern
  - Detect edges
  - Calculate the data dependent jitter (DDJ), similar to what was previously described.
  - Remove the DDJ from both edges.
  - For all pulses in the entire record, remove DDJ to leave only the uncorrelated component.

Computation of UPW-TJ/UPW-UDJDD

○ Pulse Width Jitter calculation (cont'd):



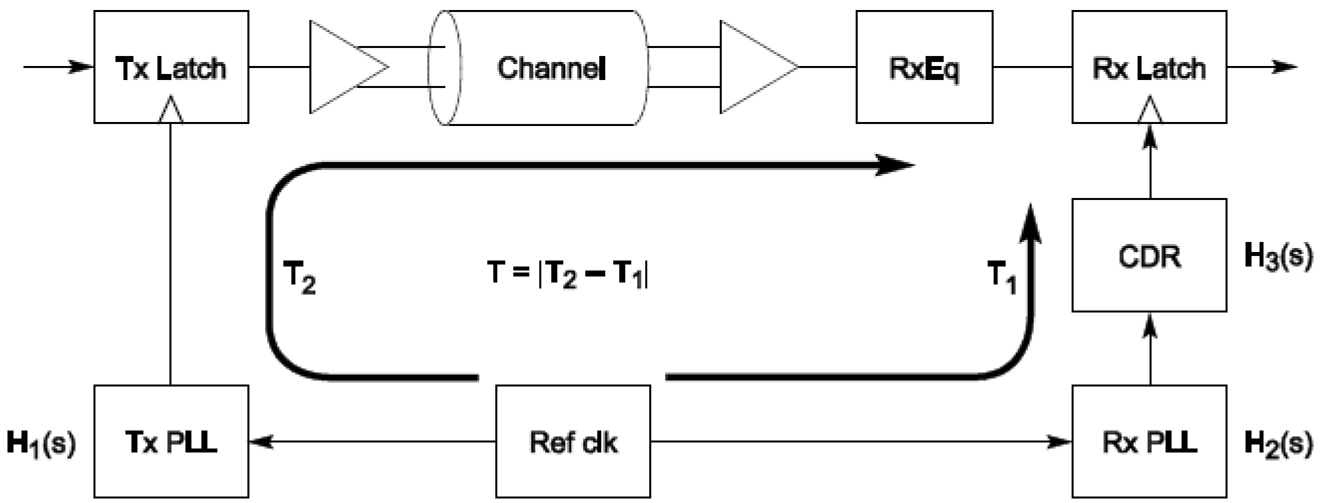
- Construct a single PDF by summing odd and even UI pulse pdfs around a normalized median.
- Convert to Q-Scale.
- Derive UPW-TJ/UPW-DJDD.

Reference Clock Jitter Measurements

- **Common Clock (CC)**
- **Independent Reference Clock (IR)**

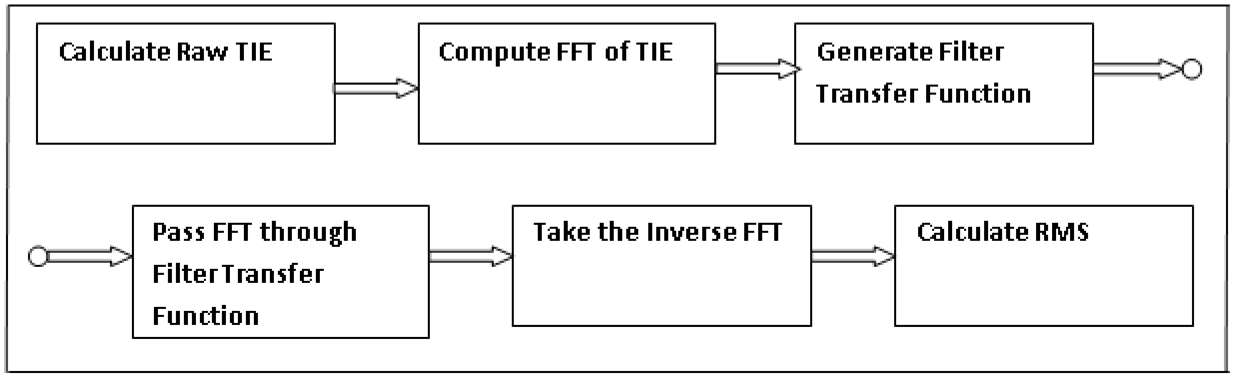
Common Clock Jitter Measurements

The common refclk architecture is described by the following diagram:



Common Clock Jitter Measurements

The measurement algorithm for RefClk Common clocked architecture:



TIE – Time Interval Error

Filter transfer function:

$H_1(s)$ → Transfer function of Tx PLL. $H_2(s)$ → Transfer function of Rx PLL. $H_3(s)$ → Transfer function of CDR.

$$H_1(s) = \left[\frac{2s\zeta_1\omega_{n1} + \omega_{n1}^2}{s^2 + 2s\zeta_1\omega_{n1} + \omega_{n1}^2}\right] \quad H_2(s) = \left[\frac{2s\zeta_2\omega_{n2} + \omega_{n2}^2}{s^2 + 2s\zeta_2\omega_{n2} + \omega_{n2}^2}\right] \quad H_3(s) = \frac{s}{s + \omega_3}$$

$$\left.\begin{array}{l} H(s) = [H_1(s)e^{-sT} - H_2(s)]H_3(s) \;\; \text{or} \\ H'(s) = [H_2(s)e^{-sT} - H_1(s)]H_3(s) \end{array}\right\} \text{Need to compute both}$$

Common Clock Jitter Measurements

In the calculation of each of $H_1(s)$ and $H_2(s)$, the bandwidth, damping, and delay are taken from publicly accessible files. The different combinations of bandwidth, delay and damping as described in the standards, for 8G/16G for e.g., is shown in Figure below:

PLL #1	0.01 dB peaking	2.0 dB peaking
$BW_{PLL}(min) = 2.0$ MHz	$\omega_{n1} = 0.448$ Mrad/s $\zeta_1 = 14$	$\omega_{n1} = 6.02$ Mrad/s $\zeta_1 = 0.73$
$BW_{PLL}(max) = 4.0$ MHz	$\omega_{n1} = 0.896$ Mrad/s $\zeta_1 = 14$	$\omega_{n1} = 12.04$ Mrad/s $\zeta_1 = 0.73$

PLL #2	0.01 dB peaking	1.0 dB peaking
$BW_{PLL}(min) = 2.0$ MHz	$\omega_{n2} = 0.448$ Mrad/s $\zeta_2 = 14$	$\omega_{n2} = 4.62$ Mrad/s $\zeta_2 = 1.15$
$BW_{PLL}(max) = 5.0$ MHz	$\omega_{n2} = 1.12$Mrad/s $\zeta_2 = 14$	$\omega_{n2} = 11.53$ Mrad/s $\zeta_2 = 1.15$

$BW_{CDR}(min) = 10$ MHz, 1st order

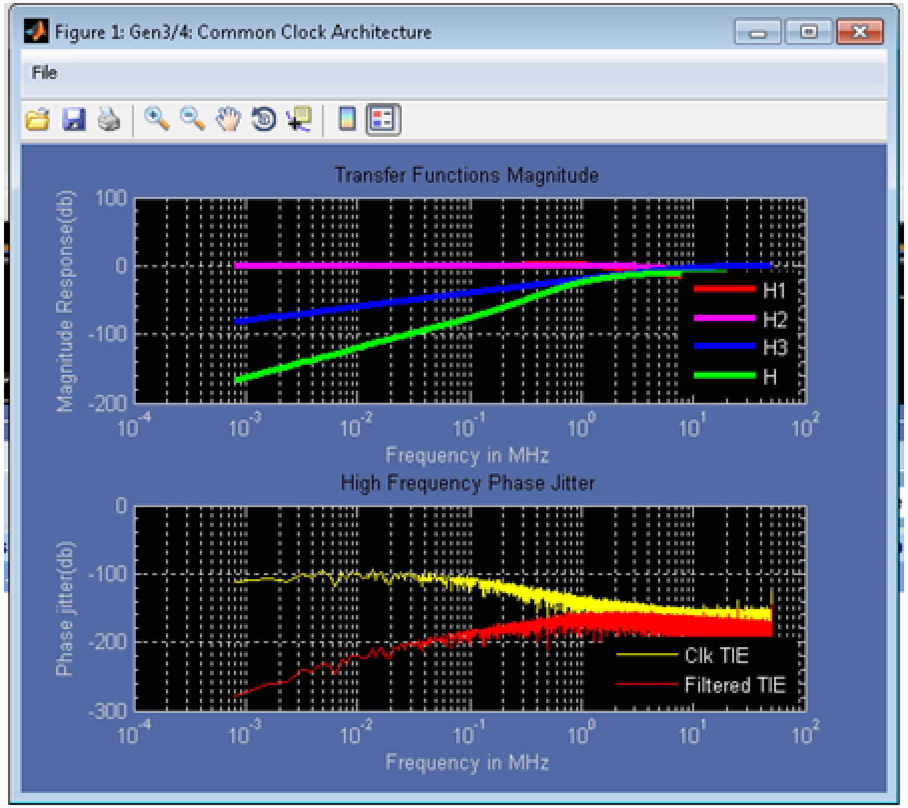
64 combinations

8.0, 16.0 GT/s

Edge filtering used to **,** minimize the measurement-induced jitter due to the finite sampling rate of the test equipment is accomplishing by band-limiting to 5GHz.

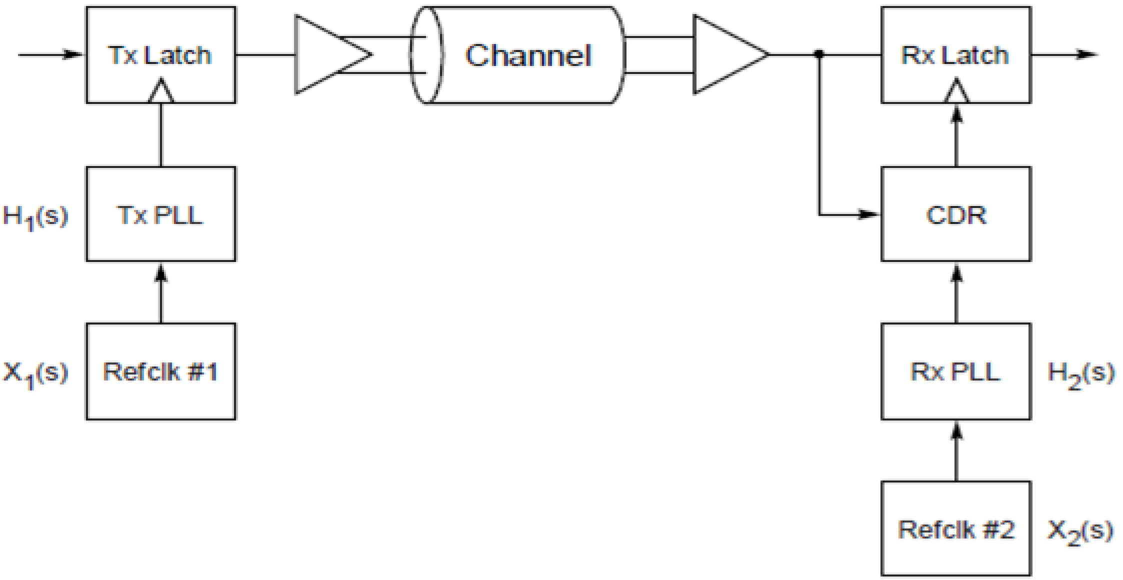
Common Clock Jitter Measurements

An example plot, of Raw TIE versus Filtered TIE is shown below.



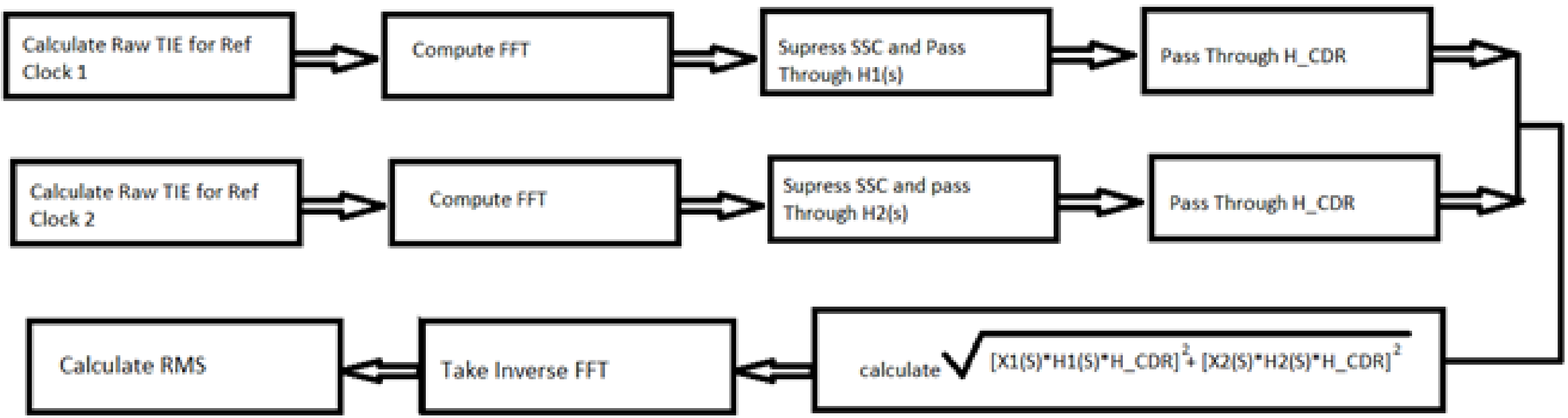
Independent Reference Clock Jitter Measurements

Independent Refclk architecture:

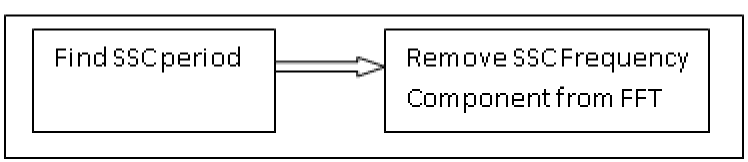


Independent Reference Clock Jitter Measurements

Measurement block diagram:



$$H_1(s)=\left[\frac{2s\zeta_1\omega_{n1}+\omega_{n1}^2}{s^2+2s\zeta_1\omega_{n1}+\omega_{n1}^2}\right] \quad H_2(s)=\left[\frac{2s\zeta_2\omega_{n2}+\omega_{n2}^2}{s^2+2s\zeta_2\omega_{n2}+\omega_{n2}^2}\right]$$



References

- Digital Communications Test and Measurement: High-Speed Physical Layer Characterization. Prentice Hall, Dennis Derickson and Marcus Muller editors
- Appendix B, Symbol Encoding, PCI Express Base Specification 4.0 0.9 spec, PCI-SIG
- Chapter 9, Electrical Sub-Block, PCI Express Base Specification 4.0 0.9 spec, PCI-SIG
- Jitter and noise measurements in the presence of crosstalk, Pavel Zivny, Tektronix
- What the Dual Dirac Model is and what it is not, Ransom Stephens, Tektronix, October 2006
- Relationship between Eye Diagram and Bathtub curves, Wavecrest corporation, 2003
- PCI Express Jitter Modeling, Revision 1.0 RD, May 2004, PCI-SIG
- Tektronix White paper for Measuring the Deterministic Jitter in PCI express

Thank you for attending the PCI-SIG Developers Conference 2017.

For more information please go to www.pcisig.com